

quinn emanuel trial lawyers | new york

51 Madison Avenue, 22nd Floor, New York, New York 10010-1601 | TEL (212) 849-7000 FAX (212) 849-7100

WRITER'S DIRECT DIAL NO.
(212) 849-7345

WRITER'S EMAIL ADDRESS
danbrockett@quinnemanuel.com

July 3, 2024

By ECF

Honorable Lewis A. Kaplan
United States District Judge
Southern District of New York
Daniel Patrick Moynihan
United States Courthouse
500 Pearl Street
Court Room 21B
New York, NY 10007-1312

Re: *Edmar Financial Company, LLC et al v. Currenex, Inc. et al*, Case No. 21-cv-06598

Dear Judge Kaplan:

Pursuant to the Court's Individual Rules, we write respectfully on behalf of Plaintiffs to request the Court direct Defendant Currenex, Inc. to produce the source code that contains the matching algorithm used on the Currenex platform, and documents related thereto. The parties have met and conferred extensively but have reached an impasse.

Background

Plaintiffs represent a putative class of traders who used Currenex's trading platform. Users of the platform submit bids and offers, and the platform matches buyers and sellers accordingly. To do this, the Currenex platform uses a "matching algorithm"—i.e., the set of instructions used to match bids and offers to execute trades, including the instructions about what to do in the event that multiple bids or multiple offers are for the same price. One possible rule for handling ties is the "first-in, first out," or "FIFO" rule, which dictates that, for two equally priced orders, the platform will give the business to the customer that submitted its price first. *See* Compl. ¶ 6 (ECF No. 96).

The crux of Plaintiffs' complaint is that, while Currenex was publicly representing that it was using the FIFO rule, in reality Currenex secretly was giving super-priority privileges to its cronies. Plaintiffs allege these privileges allowed Defendants HC Tech, Goldman Sachs, and

quinn emanuel urquhart & sullivan, llp

LOS ANGELES | NEW YORK | SAN FRANCISCO | SILICON VALLEY | CHICAGO | WASHINGTON, DC | HOUSTON | SEATTLE
LONDON | TOKYO | MANNHEIM | MOSCOW | HAMBURG | PARIS | MUNICH | SYDNEY | HONG KONG | BRUSSELS | ZURICH

State Street Bank to jump in line and consummate trades without having to enter a competitive quote.

This means that the matching algorithm is what this case is all about. By extension, this means the *source code* is what this case is all about. It is the source code that would reflect how Currenex was *actually* matching buyers and sellers on its platform. Accordingly, Plaintiffs served Currenex with a document request (Request No. 49) seeking:

- (a) the Source Code;
- (b) all changes to that Source Code (including all configuration files, configuration history, version control history, and log files of all priority changes made on the matching engine);
- (c) Documents used to request, order, or specify changes in the operation of the matching and Tiebreaking Rules by any method . . . ; and
- (d) data dictionaries and instruction manuals that describe the operation and data used by that Source Code

Currenex objected to the production of the source code and related materials on a litany of grounds, including burden. During meet-and-confers that occurred over several months, Currenex asserted it would not be necessary to produce the source code because it would produce data sufficient for the matching algorithm to be recreated, without the source code itself. Relying on that representation, Plaintiffs agreed to hold Request No. 49 in abeyance pending Currenex's production, and Plaintiffs' review, of a one-week sample of transactional data.

But Currenex only completed its production of that sample data on May 23, 2023. Plaintiffs quickly analyzed the production and determined it was nowhere close to a reasonable substitute for the actual matching algorithm. On June 5, Plaintiffs thus demanded that Currenex immediately comply with Request No. 49. Currenex ignored the request, despite multiple follow-ups. On June 29, Currenex said—despite already having a month to consider—it would be unable to respond to further conferral requests until mid-July. Plaintiffs gave Currenex a hard deadline to commit to fully producing responsive materials by July 2. On July 2, Currenex informed Plaintiffs that it would “engage in reasonable efforts to produce the source code” only if Plaintiffs agreed to produce their own algorithms and trading methodologies to Defendants. On July 3, Plaintiffs rejected this attempt to horse-trade and told Currenex that Plaintiffs would be seeking the Court's assistance. This motion followed.

Discussion

Under Federal Rule of Civil Procedure 26, “parties may obtain discovery regarding any nonprivileged matter that is relevant to any party's claim or defense and proportional to the needs of the case.” Fed. R. Civ. P. 26(b)(1). “Relevance has been construed broadly to encompass any matter that bears on, or that reasonably could lead to other matter that could bear on, any issue that is or may be in the case.” *In re LIBOR-Based Fin. Instruments Antitrust Litig.*, 2023 WL 2871090, at *5 (S.D.N.Y. Apr. 10, 2023) (emphasis added and internal citation

omitted). After relevance is established, the party objecting to the discovery bears the burden of showing why the discovery should be denied. *See, e.g., Alzheimer's Found. of Am., Inc. v. Alzheimer's Disease & Related Disorders Ass'n, Inc.*, 2015 WL 4033019, at *14 (S.D.N.Y. June 26, 2015). "Specifically, the resisting party must show how, despite the broad and liberal construction afforded the federal discovery rules, each request is not relevant, or how each request is overly broad, burdensome, or oppressive, by submitting affidavits or offering evidence revealing the nature of the burden." *Id.*; *see also Hicks v. Leslie Feely Fine Art, LLC*, 2021 WL 3617208, at *4 (S.D.N.Y. Aug. 13, 2021) (same).

The source code could not be more central to this case. This case is about whether Currenex's public proclamations of how the matching algorithm worked fairly portrayed how the matching algorithm actually worked.¹ *The source code is the matching algorithm.* Accordingly, not only is it relevant, but its production is indisputably proportional to the needs of the case. Such conclusion is further confirmed by Currenex's complete failure to buttress its boilerplate burden objections during the conferral process. *See Hicks*, 2021 WL 3617208, at *4.

Courts in this district routinely order the production of source code where, as here, it goes to a core issue in the case. *See, e.g., Liveperson, Inc. v. 24/7 Customer, Inc.*, ECF No. 197, at 8 (S.D.N.Y. June 22, 2016) (directing defendant in trade secret case to produce source code within 21 days); *Alexander Interactive, Inc. v. Adorama, Inc.*, 2013 WL 6283511 at *4-5 (S.D.N.Y. Dec. 4, 2013) (ordering inspection of source code and "commit logs" showing details of changes made to source code in case alleging defendant copied source code from plaintiff); *see also Dynamic Microprocessor Assocs. v. EKD Computer Sales*, 919 F. Supp. 101 (E.D.N.Y. 1996) (directing production of source codes for software in copyright infringement action).

Currenex's only real response is to suggest that the production of the matching algorithm (the source code) is not needed because it can be *inferred* somehow from the trade data it has produced. It cannot be that because Plaintiffs *might* be able to *guess* at the truth, that Currenex is entitled to withhold discovery directly showing the truth.

Consider a stylized hypothetical where the key factual issue is the exact terms of a mathematical formula. In such a case, of course the defendant should produce all documents regarding the formula, including what the formula is. Such is so obviously correct it is difficult to fathom why Currenex feels it appropriate to suggest that, instead, a long list of numerical *outputs* should be produced, from which the formula can supposedly be reverse-engineered.

This would be true even if the mathematical formula were simple. If it were known that for input "1" the secret formula output "2," and for input "2" the secret formula output "4," one might venture a *first guess* the formula is "2 times the input." But then again, maybe the secret formula is more complex. Or maybe it operates on a series of if-then functions that would be overlooked. Or maybe it changed across time. Thus, even in this simple scenario a plaintiff

¹ *See, e.g.,* Compl. ¶¶ 1-21, 79-125, 240; *see also* ¶ 176 ("Under a regime where FIFO was the only tiebreaking rule, the Trading Defendants would be matched by the algorithm far less often than they were."); ¶ 222 ("The Currenex Platform's algorithm for assigning trades between customers and liquidity providers is opaque and nonpublic.").

should not be forced to infer that a given set of input-output pairs mean the secret formula might be one thing, versus another.

So, too, here, all Currenex has produced is transactional data. But the transactional data does not reveal the inner mechanics of how that match was chosen. Those mechanics are found only in the source code and related materials. Plaintiffs here are entitled to the formula itself—the matching algorithm, by way of the source code—which is, in fact, millions of times stronger than in the simple scenario above.

This is because, *first*, of the enormity of the task Currenex would place upon Plaintiff. This is *not* a simple formula, with a simple list of inputs and outputs. Currenex produced *over 864 million* rows of data for just a *one-week* sample, for *one currency pair*. Both the risk of wasting time, and the risk of errors, are heightened when trying to reverse-engineer across such a massive database.

Second, the source code and related materials are needed even more here than in the stylized example because Currenex has previewed it will not even be able to give us the data we are supposed to use to formulate our guesses for a large portion of the class period.

Third, the source code and related materials are needed even more here than in the stylized example because, even if Currenex was functioning above-board, there are multiple variables in play. For instance, there are over a dozen different order types. There may be special cases relating to credit limits, whitelists, and blacklists that go into who can be matched with who. Orders may also have been matched against Currenex’s anonymous “FX Trades” ECN, which in turn had its own rules. Currenex may well have tweaked its matching algorithm regularly. This all increases the complexity of reverse-engineering, and decreases the confidence that any such reverse-engineering is accurately re-creating what the underlying algorithm was actually doing.

Finally, again, all the above would be true even if Currenex was doing everything above-board. But the crux of the case is that Plaintiffs allege the exact opposite; the core factual issue in the case is whether there were *secret rules* in the matching algorithm. This possibility not only adds additional variables and thus more burden and more improbability to the guessing task, but demonstrates the fundamental folly of Currenex’s reverse-engineering proposal. Presume Plaintiffs do manage to reverse-engineer and conclude that there was a secret rule in place, just as alleged. Currenex would, of course, attack our methodology, logic, and inferences. To subject Plaintiffs to the risk of mistakes—or even the risk of us being *accused* of making mistaken inferences—is unfair, and would waste our time, the Court’s time, and the jury’s time.

This final point thus takes the argument full circle: This case is about whether the matching algorithm was accurately represented. In such a case, it is plainly proportional to the needs of the case for Plaintiffs to be given access to the matching algorithm itself. That way there will be far less disputes—if any—over what the algorithm *is*, leaving the Court and jury to focus instead on whether the algorithm was *materially misrepresented*. Unless Currenex will give a detailed stipulation to the factual presence of the alleged features, it should not be allowed to hide the ball by refusing to produce the source code materials.

Respectfully submitted,

/s/ Daniel L. Brockett

Daniel L. Brockett

cc: Counsel of Record